AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1.   (Currently amended) A system that facilitates maintaining an item, comprising:

a first data store that stores the item in an active state;

a second data store that stores the item in an archived state; and

an inference system that inferentially determines whether to store the item in an active or archived state based at least in part upon information related to at least one of: a property of the item, a property of a user and extrinsic data, wherein the system stores the item in an archived state after a first access if the inference system determines that the user is likely to access the item only once.

~~a cost benefit analysis comparing an overhead cost attributable to storing the item in the active state to a benefit associated with storing the item in the active state, the benefit determined at least in part by calculating a probability a user will access the item in the future; and~~

~~an assessment of the item's worth relative to other items actively stored, wherein the item is stored as active according to the space available in the first data store, the cost benefit analysis, and the assessment of the item's relative worth.~~

2.   (Previously presented) The system of claim 1 further comprising:

a property log that stores user preferences gathered from implicit evidence obtained by monitoring user activity, the inference system consulting the property log when making an inferential determination.

3-4.   (Canceled).

2

5.   (Currently amended) The system of claim [[4]]1, the inference system utilizing a value density of the item to determine whether to store the item in an active or archived state perform the cost-benefit analysis.

6.   (Original) The system of claim 5 wherein the value density can be stated as the probability of user access given evidence divided by the size of the item and which can be defined as:

$$value\ density = \frac{p\ (access\,|\,\mathrm{E})}{item\ size}$$

7-9.   (Canceled).

10.   (Previously presented) The system of claim 1 further comprising:
a learning system that acts upon the inference system and modifies inferences made thereby based upon at least one of: a property of the item, a property of a user, extrinsic data, a determined probability and a value density.

11.   (Currently amended) A utility based item archiving system comprising:
means for storing an item in fast memory or archival memory;
means for storing an item in archival memory, the archival memory being slower than the fast memory;
means for determining a utility of the item, the utility determined at least in part by calculating a probability a user would prefer the item stored in fast memory;
means for determining one or more costs associated with storing the item in fast memory, at least one cost relating to the size of the item;
means for determining one or more gains associated with storing the item in fast memory, at least one gain relating to the utility of the item;
means for determining a relative utility of the item, the relative utility determined by comparing the item's utility to the utility of at least one other item; and

3

09/894,392   MS171134.01/MSFTP214US

means for inferring whether to store the item as active in fast memory or to archive the item in archival memory based upon size of the fast memory, the one or more costs, the one or more gains, and ~~the determined relative utility of the item~~whether the item is a one-shot item.

12.   (Canceled).

13.   (Previously presented) The system of claim 11 being temporally sensitive such that the determined utility of the item and storage inferences drawn therefrom are continually updated over time.

14.   (Canceled).

15.   (Previously presented) The system of claim 13 wherein the means for inferring determines a value density that is a measure of item utility versus size.

16.   (Previously presented) The system of claim 15 wherein the means for inferring employs a knapsack packing analysis to determine how to store the item.

17.   (Original) The system of claim 16 wherein, when more than one item exists, the knapsack packing analysis considers respective value densities of items to determine which items to store as active and which items to archive.

18.   (Currently amended) A method for determining how to store items comprising:
        determining respective item utilities through probabilistic computations;
        determining respective value densities of the items based upon determined probabilities of future item access and item size;
        ~~assessing a worth of an item relative to at least one other item utilizing the determined utilities and value densities;~~ and
        inferring whether to store actively ~~in fast memory~~, archive or discard items based upon the determined probabilities, value densities and whether the items are likely to be accessed only once.

4

~~space available in fast memory and the determined relative worth.~~

19.   (Previously presented) The method of claim 18 further comprising:

updating determined item utilities over time;

determining modern value densities based upon updated item utilities; and

making contemporary inferences from at least one of: the temporally adjusted

probabilities and value densities.

20.   (Currently amended) The method of claim 19 further ~~including~~ comprising:

mitigating inefficient use of active space by applying a knapsack computation to the

modern item value densities.

21.   (Currently amended) A method of streamlining actively stored items comprising:

determining item utilities and value densities;

ordering items according to their utilities and value densities;

storing items in an active item store having fast memory according to their utilities and

value densities as space permits;

removing <u>one or more</u> items from the active item store to make space available for an

item with a higher utility and value density, wherein removing <u>the one or more</u> items comprises

archiving the items in a slower data store when at least one of: determined item utilities and

value densities fall below a predetermined threshold<u>; and</u>

<u>removing one-shot items from the active item store after the one-shot items are accessed</u>

<u>a first time</u>.

22.   (Canceled).

23.   (Currently amended) A computer-readable medium storing computer-executable

instructions for performing the method of claim [[18]]<u>21</u>.

24.   (Currently amended) A system to infer how to store an item, comprising:

5

a probability component to determine a probability that the item will be accessed at a later time;

a cost-benefit component to determine a value density of the item as a function of the determined probability[[,]] and size of the item; and

an inference system to infer whether to store the item in an active fast memory or archive the item in a slower store based upon the determined value density, space available in the fast memory[[,]] and the value density of at least one other item, wherein the system archives the item after a first access if the inference system infers that the item will be accessed only once.

25.   (Previously presented) The system of claim 24 wherein the probability component determines the probability based upon at least one of: information related to a property of the item, a property of a user and extrinsic data.

26.   (Currently amended) The system of claim 25 further comprising:

a property log operative to store information regarding at least one of: a property of the item, a property of the user and extrinsic data.

27.   (Original) The system of claim 26 being temporally sensitive such that information stored in the property log is updated over time.

28.   (Currently amended) The system of claim 27 being temporally sensitive such that at least one of: the determined probability, value density and inferences drawn therefrom [[is]]are updated over time.

29.   (Previously presented) The system of claim 28 further comprising:

a learning system to learn how to adjust the inference system based upon at least one of: updated log information, determined probabilities and value densities.

30.   (Original) The system of claim 24 further comprising:

an interactive user interface (UI).

6

31.   (Currently amended) The system of claim 30 wherein conditions are utilized by the probability component and inference system, the UI including a selection element ~~operative~~ to allow a condition to be enabled/disabled.

32.   (Currently amended) The system of claim 31 wherein the UI further ~~includes~~ comprises an entry element ~~operative~~ to allow a condition to be configured.

33.   (Currently amended) A system to determine which of a plurality of items to store actively comprising:

       a probability component to determine respective probabilities that the plurality of items subsequently will be accessed;

       a cost-benefit component to determine respective value densities as a function of the respective probabilities and sizes of the plurality of items;

       an inference system to infer whether the plurality of items should be actively stored in fast memory based upon the respective value densities; and

       an optimization component to determine which of the plurality of items to remove from being actively stored and place in an archival store based upon a comparison of relative value densities of the items, ~~and~~ an amount of active space available and whether the items are likely to be accessed more than once.

34.   (Currently amended) The system of claim 33 wherein the probability component determines the respective probabilities of the plurality of items based upon at least one of: respective properties of the items, a property of a user and extrinsic data.

35.   (Currently amended) The system of claim 34 further comprising:

       a property log operative to store information regarding at least one of respective properties of the plurality of items, a property of the user and extrinsic data.

36.   (Original) The system of claim 35 being temporally sensitive such that information stored in the property log is updated over time.

7

37.   (Previously presented) The system of claim 36 being temporally sensitive such that at least one of determined probabilities, value densities and inferences drawn therefrom are updated over time.

38.   (Currently amended) The system of claim 37 further comprising:
        a learning system ~~adapted~~ to learn how to adjust the inference system based upon at least one of: updated log information, determined probabilities and value densities.

39.   (Currently amended) A method to determine which of a plurality of items to store actively comprising:
        determining respective probabilities that the plurality of items will be accessed in the future;
        determining respective value densities as a function of the respective probabilities and sizes of the plurality of items;
        ordering the plurality of items according to their value densities;
        inferring whether the plurality of items should be actively stored in fast memory based upon at least one of: respective probabilities and value densities;
        determining which of the plurality of items to store in active space based upon the order of the value densities ~~of the items~~ and an amount of active space available; and
        determining which of the plurality of items to remove from active space based upon the order of the value densities ~~of the items~~ and the amount of active space available, and wherein items that are likely to be accessed only once are removed from active space after they are accessed once.

40.   (Currently amended) The method of claim 39 further comprising:
        determining the respective probabilities ~~of the items~~ based upon at least one of: respective properties of the items, a property of a user and extrinsic data.

41.   (Previously presented) A computer-readable medium storing computer-executable instructions for performing the method of claim 39.

8

42.   (Previously presented) A system to infer whether an item will be accessed only once or more than once comprising:

an inference system to infer whether an item will be accessed only once or more than once and to discard or move one or more items likely to be accessed only once to a slower memory store after the one or more items are accessed once, the inference based on a comparison of properties of the item to properties of other items that have been accessed only once, the inference system analyzing properties of a user and extrinsic data; the inference system having:

a probability component to determine the probability that the item will be accessed only once; and

a cost-benefit component to determine a value density of the item as a function of the determined probability, the inference system comparing at least one of: the determined probability and value density to probabilities and value densities of items that have been accessed only once to infer whether the item is likely to be accessed only once.

43.   (Previously presented) A method to infer whether an item will be accessed once or more than once comprising:

comparing properties of the item to properties of other items that have been accessed once;

analyzing properties of a user and extrinsic data;

determining a probability that the item will be accessed only once;

determining a value density of the item as a function of the determined probability and size of the item;

comparing the determined probability and value density to probabilities and value densities of items that have been accessed once; and

removing the item from fast memory after the item has been accessed if at least one of the determined probability and value density of the item are similar to at least one of probabilities and value densities of other items that have been accessed once.

44.   (Previously presented) A computer-readable medium storing computer-executable instructions for performing the method of claim 43.

45.   (Currently amended) An interactive user interface (UI) to display a condition that affects how a decision is made regarding the storage of an item comprising:

a selection element to allow a condition to be enabled/disabled; and

an entry element to permit a condition to be configured, wherein the condition relates to a probabilistic analysis for determining whether to store the item in a fast memory or in an archival memory, the probabilistic analysis based at least in part on the item's worth relative to other items to be stored and size of the fast memory[[,]] and wherein the item's worth is determined at least in part by calculating a probability a user will access the item at some later time, the item's worth being dependent at least in part on whether the item is a one-shot item.